

JAMES Y. PAK (Bar No. 304563)  
SKADDEN, ARPS, SLATE, MEAGHER & FLOM LLP  
525 University Ave., Suite 1400  
Palo Alto, California 94301-1908  
Telephone: (650) 470-4500  
Facsimile: (650) 470-4570  
james.pak@skadden.com

P. ANTHONY SAMMI (*pro hac vice*)  
DOUGLAS R. NEMEC (*pro hac vice*)  
EDWARD L. TULIN (*pro hac vice*)  
SKADDEN, ARPS, SLATE, MEAGHER & FLOM LLP  
Four Times Square  
New York, NY 10036-6522  
Telephone: (212) 735-3000  
Facsimile: (212) 735-2000/1  
anthony.sammi@skadden.com  
douglas.nemec@skadden.com  
edward.tulin@skadden.com

*Attorneys for Defendants*  
INSIDE SECURE, S.A. AND  
INSIDE SECURE CORP.

UNITED STATES DISTRICT COURT  
NORTHERN DISTRICT OF CALIFORNIA  
OAKLAND DIVISION

PACE ANTI-PIRACY, INC. a California  
corporation,  
  
Plaintiff,  
  
v.  
  
INSIDE SECURE, a French legal entity,  
and INSIDE SECURE CORP., a Delaware  
corporation,  
  
Defendants.

Case No.: 4:17-cv-05860-HSG

**DEFENDANTS INSIDE SECURE S.A. AND  
INSIDE SECURE CORP.'S NOTICE OF  
MOTION AND MOTION TO DISMISS  
COMPLAINT PURSUANT TO FEDERAL  
RULE OF CIVIL PROCEDURE 12(b)(6);  
MEMORANDUM OF POINTS &  
AUTHORITIES IN SUPPORT THEREOF**

**Hearing Date: Mar. 22, 2018  
Time: 2:00 p.m.**

1 TO ALL PARTIES AND THEIR ATTORNEYS OF RECORD:

2 PLEASE TAKE NOTICE that on March 22, 2018 at 2:00 p.m., or as soon thereafter as the  
3 matter may be heard in the above-entitled court, located at 1301 Clay Street, Oakland, CA 94612,  
4 Defendants Inside Secure S.A. and Inside Secure Corp. ("Defendants") will and do hereby move  
5 the Court for dismissal of the Complaint pursuant to Rule 12(b)(6) of the Federal Rules of Civil  
6 Procedure.

7 Plaintiff PACE Anti-Piracy Inc. has failed to state a claim for which relief may be granted  
8 because the asserted claims of U.S. Patent No. 6,880,149 are invalid under 35 U.S.C. § 101, and  
9 therefore the Complaint should be dismissed. This Motion is based upon this Notice of Motion and  
10 Motion, the Memorandum of Points and Authorities attached hereto, the Complaint, the pleadings  
11 and papers on file in this action, and upon such other and further evidence and argument as may be  
12 submitted at or before the hearing on this matter.

13

14 DATED: December 26, 2017

Respectfully submitted,

15

SKADDEN, ARPS, SLATE, MEAGHER & FLOM LLP

16

By: /s/ James Y. Pak

17

James Y. Pak

18

Attorney for Defendants

19

20

21

22

23

24

25

26

27

28

## TABLE OF CONTENTS

		<u>Page</u>
1		
2		
3		
4	TABLE OF AUTHORITIES .....	ii
5	MEMORANDUM OF POINTS AND AUTHORITIES .....	1
6	I. STATEMENT OF ISSUES TO BE DECIDED .....	1
7	II. PRELIMINARY STATEMENT .....	1
8	III. FACTUAL BACKGROUND .....	2
9	A. The Parties .....	2
10	B. The '149 Patent.....	3
11	IV. LEGAL STANDARDS .....	6
12	A. Legal Standard Under Rule 12(b)(6) .....	6
13	B. Legal Standard For Patent Eligibility Under Section 101 Of The Patent Act .....	7
14	V. THE '149 PATENT CLAIMS UNPATENTABLE SUBJECT MATTER.....	8
15	A. The Asserted Claims Fail To Satisfy Step 1 Of The <i>Alice</i> Test Because They Are Directed To An Abstract Idea .....	8
16	1. The Asserted Claims Are Merely Directed To Using A Conventional Algorithm And Comparing Two Values To See If They Match .....	8
17	2. The Asserted Claims Fail The Pen-And-Paper Test .....	10
18	3. The Asserted Claims Are Closely Analogous To Claims That Have Been Recently Invalidated Under Section 101 .....	11
19	B. The Asserted Claims Fail to Satisfy Step 2 Of The <i>Alice</i> Test Because They Do Not Contain An Inventive Concept.....	15
20		
21	VI. CONCLUSION.....	18
22		
23		
24		
25		
26		
27		
28		

**TABLE OF AUTHORITIES**

**Cases**

<i>Alice Corp. Pty. Ltd. v. CLS Bank International</i> , 134 S. Ct. 2347 (2014).....	1, 7, 9, 15
<i>Ancora Technologies, Inc. v. HTC America, Inc.</i> , No. C16-1919 RAJ, 2017 WL 6389329 (W.D. Wash. Dec. 14, 2017) .....	2, 16
<i>Asghari-Kamrani v. United Services Automobile Ass'n</i> , No. 2:15cv478, 2016 WL 3670804 (E.D. Va. July 5, 2016) .....	14
<i>Bancorp Services, L.L.C. v. Sun Life Assurance Co. of Canada (U.S.)</i> , 687 F.3d 1266 .....	18
<i>Bilski v. Kappos</i> , 561 U.S. 593 (2010).....	6
<i>Blue Spike, LLC v. Google Inc.</i> , No. 14-cv-01650-YGR, 2015 WL 5260506 (N.D. Cal. Sept. 8, 2015), <i>aff'd</i> , 669 F. App'x 575 (Fed. Cir. 2016), <i>cert. denied</i> , 137 S. Ct. 2246 (2017).....	7, 14
<i>buySAFE, Inc. v. Google, Inc.</i> , 765 F.3d 1350 (Fed. Cir. 2014).....	6
<i>Clarilogic Inc. v. FormFree Holdings Corp.</i> , 681 F. App'x 950 (Fed. Cir. 2017) .....	9
<i>Content Extraction &amp; Transmission LLC v. Wells Fargo Bank National Ass'n</i> , 776 F.3d 1343 (Fed. Cir. 2014).....	7
<i>CyberSource Corp. v. Retail Decisions, Inc.</i> , 654 F.3d 1366 (Fed. Cir. 2011).....	10, 18
<i>FairWarning IP, LLC v. Iatric Systems, Inc.</i> , 839 F.3d 1089 (Fed. Cir. 2016).....	13
<i>Intellectual Ventures I LLC v. Erie Indemnity Co.</i> , No. 2017-1147, 2017 WL 5041460 (Fed. Cir. Nov. 3, 2017) .....	2, 11
<i>Intellectual Ventures I LLC v. Erie Indemnity Co.</i> , 200 F. Supp. 3d 565 (W.D. Pa. 2016), <i>aff'd</i> , No. 2017-1147, 2017 WL 5041460 (Fed. Cir. Nov. 3, 2017) .....	3, 11, 12
<i>Intellectual Ventures I LLC v. Capital One Financial Corp.</i> , 850 F.3d 1332 (Fed. Cir. 2017).....	15
<i>Intellectual Ventures I LLC v. Symantec Corp.</i> , 838 F.3d 1307 (Fed. Cir. 2016).....	12, 16
<i>Mayo Collaborative Services v. Prometheus Laboratories, Inc.</i> , 132 S. Ct. 1289 (2012).....	8

1	<i>OIP Technologies, Inc. v. Amazon.com, Inc.</i> ,	
2	788 F.3d 1359 (Fed. Cir. 2015).....	7
3	<i>Open Text S.A. v. Box, Inc.</i> ,	
4	78 F. Supp. 3d 1043 (N.D. Cal. 2015) .....	8
5	<i>OpenTV, Inc. v. Apple, Inc.</i> ,	
6	Case No. 14-cv-01622-HSG, 2015 WL 1535328 (N.D. Cal. Apr. 6, 2015).....	10, 13, 14, 17
7	<i>Pace Anti-Piracy, Inc. v. Inside Secure et al.</i> ,	
8	Case No. 5:17-cv-06744-BLF (N.D. Cal.).....	3
9	<i>Papst Licensing GmbH &amp; Co. KG v. Xilinx Inc.</i> ,	
10	193 F. Supp. 3d 1069 (N.D. Cal. 2016), <i>aff'd</i> , 684 F. App'x 971	
11	(Fed. Cir. 2017).....	8, 9, 10, 16, 17
12	<i>Planet Bingo, LLC v. VKGS LLC</i> ,	
13	576 F. App'x 1005 (Fed. Cir. 2014).....	13
14	<i>Protegrity USA, Inc. v. Netskope, Inc.</i> ,	
15	No. 15-cv-02515-YGR, 2015 WL 6126599 (N.D. Cal. Oct. 19, 2015) .....	16, 17
16	<i>Ultramercial, Inc. v. Hulu</i> ,	
17	772 F.3d 709 (Fed. Cir. 2014), <i>cert. denied</i> , 135 S. Ct. 2907 (2015) .....	3, 6, 15, 16, 17
18	<i>Vehicle Intelligence &amp; Safety LLC v. Mercedes-Benz USA, LLC</i> ,	
19	635 F. App'x 914 (Fed. Cir. 2015).....	13
20	<b>Statutes</b>	
21	35 U.S.C. § 101.....	7
22	<b>Rules</b>	
23	Fed. R. Civ. P. 8(a)(2).....	6
24	Fed. R. Civ. P. 12(b)(6).....	6

**MEMORANDUM OF POINTS AND AUTHORITIES**

**I. STATEMENT OF ISSUES TO BE DECIDED**

Whether Plaintiff PACE Anti-Piracy Inc. ("Plaintiff") has failed to state a claim for which relief may be granted because the asserted claims of U.S. Patent No. 6,880,149 ("the '149 patent") are invalid under 35 U.S.C. § 101.

**II. PRELIMINARY STATEMENT**

With its 2014 decision in *Alice Corp. Pty. Ltd. v. CLS Bank Int'l*, 134 S. Ct. 2347 (2014), the Supreme Court of the United States made arguably the most dramatic change to patent law in modern history, holding that patent claims directed to software and "business methods" are, by and large, unpatentable under 35 U.S.C. § 101. What is more, *Alice* confirmed that whether a patent claim is directed to patentable subject matter is a question of law that courts may resolve at the pleading stage, without the need for discovery, expert opinion, or claim construction. Applying *Alice* and its progeny, district courts have granted—and the U.S. Court of Appeals for the Federal Circuit has affirmed—motions to dismiss patent cases at unprecedented rates. As explained herein, Plaintiff's claims should likewise be dismissed.

The method claimed in the '149 Patent consists of essentially four generic steps: (1) preparing a software program to perform a mathematical calculation; (2) performing that calculation; (3) removing any evidence that such calculation has been performed; and (4) comparing the result of that calculation to a known value to see if the software has been modified. The Federal Circuit has propounded a variety of tests to assess whether claims cover patentable subject matter. The '149 Patent claims fail all of them, because the foregoing method *can* be performed in the human mind and with pen and paper; it does *not* improve the functioning of a computer; it is *not* unique to the Internet; it does *not* involve a machine, nor does it transform matter.

Indeed, the '149 Patent, which issued nearly a decade before the Supreme Court's sea-changing decision, is precisely the sort of pure software patent that *Alice* intended to abolish. It is directed to the abstract idea of determining whether software has been modified. And rather than

including an inventive concept, the asserted claims instead rely on well-known mathematical formulas (known as "checksums"), fundamental programming concepts (deleting certain portions of computer code), and basic observation (comparing two values to determine if they match) to achieve the claimed result. This abstract idea has no concrete or tangible form, with the claims of the '149 patent instead focused exclusively on a desired result (detection of software tampering) using conventional mathematical and observational principles. It is no surprise, then, that these method claims are closely analogous to those that both district courts and the Federal Circuit have consistently determined are invalid in the wake of *Alice*. Indeed, less than two months ago, the Federal Circuit affirmed the unpatentability of claims directed to computer-based comparisons that used *the same checksum-based method* as the '149 Patent. *Intellectual Ventures I LLC v. Erie Indemnity Co.*, No. 2017-1147, 2017 WL 5041460 (Fed. Cir. Nov. 3, 2017). And less than two weeks ago, a district court invalidated a patent that was directed to *the same abstract verification concept to combat the same sort of software piracy* as the '149 Patent. *Ancora Techs., Inc. v. HTC Am., Inc.*, No. C16-1919 RAJ, 2017 WL 6389329 (W.D. Wash. Dec. 14, 2017).

All of the asserted claims are directed to unpatentable subject matter under 35 U.S.C. § 101. As such, Plaintiff has failed to assert a claim for relief, and the Complaint should be dismissed in its entirety.

### III. FACTUAL BACKGROUND

#### A. The Parties

Plaintiff is a privately held California corporation with its principal place of business in San Jose, California. Plaintiff "provide[s] services and software that it bundles together with its own products to provide security software to customers." (*Pace Anti-Piracy, Inc. v. Inside Secure et al.*, Case No. 5:17-cv-06744-BLF (N.D. Cal.), D.I. 30 at 2.)

Defendant Inside Secure S.A. is a publicly held French *société anonyme* with its principal place of business in Meyreuil, France. (Compl. ¶ 3.) Founded in 1994, Inside Secure S.A. is a leader in providing essential security software and other technological tools needed to protect mobile and connected device platforms. Inside Secure S.A. has five directly held subsidiaries,

1 including co-defendant Inside Secure Corp., which is a Delaware corporation with its principal  
2 place of business in San Jose, California. (Compl. ¶ 4.)

3 In 2010, Inside Secure S.A.'s predecessor-in-interest entered into a written software license  
4 agreement with Plaintiff ("the License Agreement"), under which Plaintiff received a license to the  
5 sole product accused of infringement in this litigation—the Core software. Core is an example of  
6 software that Plaintiff "bundles together with its own products" as part of a combined product.  
7 This case thus presents the highly unusual circumstance where Plaintiff is alleging infringement by  
8 a product that Plaintiff itself has licensed and used for nearly a decade.<sup>1</sup>

### 9 **B. The '149 Patent**

10 The '149 Patent was examined using a very different patentability standard than that which  
11 exists today. The '149 patent issued on April 12, 2005, nearly a decade before the Supreme Court  
12 radically altered the Section 101 calculus for computer-based subject matter. *See, e.g.,*  
13 *Ultramercial, Inc. v. Hulu*, 772 F.3d 709, 720 (Fed. Cir. 2014) (Mayer, J., concurring) (noting that  
14 patents issued prior to the Supreme Court's decisions in *Mayo* and *Alice* were examined under "an  
15 insufficiently rigorous subject matter eligibility standard"); *see also Intellectual Ventures I LLC v.*  
16 *Erie Indem. Co.*, 200 F. Supp. 3d 565, 577 (W.D. Pa. 2016) (noting that when "prosecution  
17 occurred pre-*Alice*," any reliance on PTO findings "is of limited value" when assessing patent  
18 eligibility under the current legal regime).

19 The '149 Patent is broadly directed to a generic method of determining whether software  
20 has been modified. (*See* '149 Patent at 1:7-9 (noting that "[t]he present invention relates to  
21 software piracy detection and prevention, and more particularly to detecting code modifications  
22 made to bypass anti-piracy functions").) This problem, while not unique to software, is particularly  
23 pervasive in that field, where hackers may be able to gain unauthorized access to a computer  
24 program by skirting code-based protection schemes. (*See id.* at 1:14-31.) For instance, a software  
25 program might be designed to run only a single copy on a single system, but a software hacker or  
26

---

27 <sup>1</sup> Plaintiff has also separately sued Defendants for breach of the License Agreement in  
28 a case that is pending in this judicial district. *See PACE Anti-Piracy, Inc. v. Inside Secure et al.*,  
Case No. 5:17-cv-06744-BLF (N.D. Cal.).



1 pirate may modify the software so that he can make a number of free copies of it by effectively  
2 neutralizing the portion of the code that protects against unauthorized copying. (*Id.*)

3 At the time the '149 Patent was filed, there were a number of ways to protect software from  
4 this sort of piracy, including "encryption schemes, digital signatures, and the use of license files of  
5 various types," (*id.* at 1:33-35), along with what was known as a "runtime validation process" or a  
6 checksum. (*Id.* at 3:1-3.) A checksum is simply a mathematical formula that calculates a numeric  
7 or alphanumeric value based on particular inputs. This checksum is used to "check" the integrity or  
8 validity of the input; if something in that input changes, then the checksum will change. The result  
9 of the checksum calculation can be compared with the known, correct value to see if there has been  
10 an undesired modification to the input. A hypothetical illustration is shown below:

11	Input:	Checksum Result:	Correct Checksum:	Result:
12	The United States of America	9e107d9d372bb6826bd81d3542a419d6	9e107d9d372bb6826bd81d3542a419d6	Match; statement is correct and authentic
13				
14	The united States of America	e4d909c290d0fb1ca068ffaddf22cbd0	9e107d9d372bb6826bd81d3542a419d6	No match; statement has been altered
15				

16 Even a small change to the text results in a completely different checksum, and thus makes it very  
17 easy to detect when any change has been made. This process works exactly the same way for  
18 computer code. As discussed in the specification of the '149 patent, this process was well-known  
19 in the prior art: during the execution of a computer program "a function call . . . is made to a  
20 checksum computation routine . . . to compute the checksum on the program or some portion  
21 thereof." (*Id.* at 3:12-15.) Thereafter the "results of the checksum computation" are compared with  
22 "a pre-computed checksum . . . stored within the program. If the checksums compare, then the  
23 code has not changed . . . , [but] [i]f the checksums do not compare, the application . . . does not  
24 normally execute." (*Id.* at 3:15-22.) In other words, if a hacker changes something to circumvent  
25 the scheme preventing her from gaining access, then the checksum will likewise change and detect  
26 the presence of this undesired modification.

27 The goal of a checksum in the software context is to provide a means to detect when a  
28 program has been hacked. ('149 Patent at 1:39-51.) If a computer hacker can locate a checksum,

1 then he can insert a workaround that will calculate the correct checksum even though the program  
 2 has been modified, effectively rendering the checksum meaningless. In that case, the "comparison  
 3 routine" will always "find the pre-computed checksum will be identical to the computed value,"  
 4 thus circumventing the verification step and making it appear that the software has not been  
 5 modified (though it has been). (*Id.* at 3:46-61.) Anti-piracy checksums should therefore be as  
 6 invisible as possible to hackers, and a number of conventional techniques were available to  
 7 obfuscate the fact that checksum protection had been employed in software. Those techniques and  
 8 strategies, which include placing checksums at a "number of points" in the code, and "delaying the  
 9 reaction when the comparison routine . . . returns a failure," were well known at the time the '149  
 10 Patent was filed. (*See id.* at 1:44, 4:1-8.)

11 The '149 patent claims are directed to the most basic concepts of this verification process—  
 12 using a checksum to detect whether computer code has been modified, and obfuscating the fact that  
 13 this verification process is occurring by: (1) stripping out signals that might tip a hacker off to the  
 14 presence of a checksum and (2) breaking up the checksum verification process into multiple steps.  
 15 Plaintiff has asserted eight method claims against Defendants: claims 1, 2, 3, 5, 6, 27, 30, and 31  
 16 (collectively, "the Asserted Claims"). The two asserted independent claims are reproduced below:

17 1. A method for performing runtime checksum validation of a  
 18 software program, the method comprising the steps of:

19 (a) providing a software tool as well as instructions on how to modify  
 20 the software program to submit to the tool, wherein executable code  
 21 is generated from the modified software program that includes  
 22 checksum information for the tool to use when processing the  
 23 software program;

24 (b) in response to the executable code being submitted to the tool, the  
 25 tool calculates at least one checksum, embeds the checksum in the  
 26 executable code in a location indicated by the checksum information,  
 27 and strips the checksum information from the executable code; and

28 (c) delivering the executable code as a protected software program,  
 wherein during execution, the protected software application  
 generates a new checksum and determines that the software  
 application has been modified if the new checksum fails to match the  
 embedded checksum.

27 27. A method for determining if protected software is modified, the  
 28 method comprising the steps of

(a) instructing a software developer to modify an executable version of the software to be protected by performing the steps of:

(i) exporting predefined checksum related data,

(ii) adding code to compute a new checksum at runtime,

(iii) adding code to compare the computed runtime checksum with a checksum stored in the executable,

(iv) determining that the protected software has been modified if the checksums do not compare; and

(b) wrapping the executable in a anti-piracy software wrapper, and performing the steps of:

(i) automatically detecting the exported checksum related data,

(ii) computing a checksum of the executable,

(iii) embedding the checksum in the executable; and

(iv) removing the exported checksum related symbols from the executable.

(See '149 Patent.) Those independent claims are not directed to a particular checksum, do not refer to any particular system or portion of computer architecture, and are written in exclusively functional language that relates exclusively to intangible computer code. The additional limitations of the asserted dependent claims generally fall into one of three categories: (1) specifying a particular manner of storing the checksum in claim 2; (2) specifying that multiple checksum calculations should be performed in claims 3 and 31; and (3) providing a generic means of selecting a checksum in claims 5, 6, and 30.

#### **IV. LEGAL STANDARDS**

##### **A. Legal Standard Under Rule 12(b)(6)**

A complaint must include "a short and plain statement of the claim showing that the pleader is entitled to relief." Fed. R. Civ. P. 8(a)(2). Rule 12(b)(6) requires that a complaint be dismissed if it fails "to state a claim upon which relief can be granted." Fed. R. Civ. P. 12(b)(6). In light of the Supreme Court's instruction that patent eligibility is a "threshold" issue, *Bilski v. Kappos*, 561 U.S. 593, 602 (2010), the Federal Circuit has repeatedly confirmed that disposing of patent-ineligible claims on a motion to dismiss is appropriate, because no claim for relief can be based on

such unpatentable subject matter. *See, e.g., Ultramercial*, 772 F.3d at 717; *buySAFE, Inc. v. Google, Inc.*, 765 F.3d 1350, 1352 (Fed. Cir. 2014). "Addressing 35 U.S.C. § 101 at the outset not only conserves scarce judicial resources and spares litigants the staggering costs associated with discovery and protracted claim construction litigation, it also works to stem the tide of vexatious suits brought by the owners of vague and overbroad business method patents." *OIP Techs., Inc. v. Amazon.com, Inc.*, 788 F.3d 1359, 1364 (Fed. Cir. 2015) (Mayer, J., concurring).<sup>2</sup>

### **B. Legal Standard For Patent Eligibility Under Section 101 Of The Patent Act**

The Patent Act provides that "[w]hoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title." 35 U.S.C. § 101. However, Section 101 has long been interpreted by the Supreme Court as "contain[ing] an important implicit exception: Laws of nature, natural phenomena, and abstract ideas are not patentable." *Alice*, 134 S. Ct. at 2354. "The 'abstract' ideas category embodies 'the longstanding rule that '[a]n idea of itself is not patentable.'" *Id.* at 2355 (alteration in original) (citations omitted). When considering whether subject matter falls into this category, courts "must distinguish between patents that claim the "'buildin[g] block[s]'" of human ingenuity and those that integrate the building blocks into something more." *Id.* at 2354 (citation omitted).

The Supreme Court has established a two-part test for determining whether a claim is directed to subject matter that is not eligible for patent protection. *See Alice*, 134 S. Ct. at 2355. In Step 1 of this test, the court asks whether the claims "are directed to a patent-ineligible concept[]," such as an abstract idea. *Id.* If the court answers this question in the affirmative, it then moves to Step 2 and assesses whether the patent claims include "an "'inventive concept'" that can "'transform' the . . . abstract idea into a patent-eligible invention." *Id.* at 2357 (citation omitted).

---

<sup>2</sup> It is not necessary for the Court to engage in any formal claim construction to resolve this motion. *See, e.g., Content Extraction & Transmission LLC v. Wells Fargo Bank Nat'l Ass'n*, 776 F.3d 1343, 1349 (Fed. Cir. 2014) ("[C]laim construction is not an inviolable prerequisite to a validity determination under [Section] 101."); *Blue Spike, LLC v. Google Inc.*, No. 14-cv-01650-YGR, 2015 WL 5260506, at \*3 (N.D. Cal. Sept. 8, 2015) ("A court may decide [a motion to dismiss] prior to claim construction.").

The Court defined this inventive concept as "an element or combination of elements that is sufficient to ensure that the patent in practice amounts to significantly more than a patent upon the [ineligible concept] itself." *Id.* at 2355 (alteration in original) (citation omitted). If a claim is directed merely to the implementation of an abstract idea on computer software, or to "well-understood, routine, conventional activity" or technology, then there is no "inventive concept" sufficient to satisfy Step 2 of the *Alice* test. *Mayo Collaborative Servs. v. Prometheus Labs., Inc.*, 132 S. Ct. 1289, 1298 (2012).

When using this two-step test to "evaluat[e] whether . . . claims are directed to patent-ineligible abstract ideas, courts have generally begun by 'compar[ing] claims at issue to those claims already found to be directed to an abstract idea in previous cases.'" *Papst Licensing GmbH & Co. KG v. Xilinx Inc.*, 193 F. Supp. 3d 1069, 1081 (N.D. Cal. 2016) (citations omitted). In addition, when considering whether computer-based claims are directed to an unpatentable abstract idea, courts have typically considered whether the claims "purport to 'improve the functioning of the computer itself,'" whether the claims are directed solely to a problem that is "particular to the Internet," or whether the claims are, "in essence, directed to a mental process or a process that could be done with pen and paper." 193 F. Supp. 3d at 1081-82 (citations omitted).

## **V. THE '149 PATENT CLAIMS UNPATENTABLE SUBJECT MATTER**

### **A. The Asserted Claims Fail To Satisfy Step 1 Of The *Alice* Test Because They Are Directed To An Abstract Idea**

#### **1. The Asserted Claims Are Merely Directed To Using A Conventional Algorithm And Comparing Two Values To See If They Match**

To evaluate whether the asserted claims of the '149 Patent are impermissibly directed to an abstract idea, the Court first must "distill[] the gist of the claim[s]." *Open Text S.A. v. Box, Inc.*, 78 F. Supp. 3d 1043, 1046 (N.D. Cal. 2015). The gist of the Asserted Claims can be easily distilled: they are directed to the abstract idea of determining whether software has been modified by performing a basic comparison of a mathematical calculation to a known result. This distillation is closely tethered to the language of the claims; indeed, the preamble of asserted claim 27 provides that it is directed to "a method for determining if protected software is modified." ('149 Patent, 14:63-64.)

1 The claims address a long-standing problem (how to detect when data has been tampered  
2 with) using a long-standing solution (the comparison of a verification code or calculation to a  
3 known standard). This abstract idea is familiar to anyone who has watched an action film  
4 involving a nuclear submarine. In order to verify the authenticity of nuclear launch orders, those  
5 orders often include an embedded alphanumeric code that is compared against a known value to  
6 see if they match. That is essentially what the Asserted Claims recite, albeit with the alphanumeric  
7 verification code generated using a mathematical formula known as a checksum (which itself  
8 represents an abstract idea). As noted above, a checksum is merely an algorithm that the '149  
9 patent itself explains was in widespread use prior to the alleged invention. The heart of the  
10 independent claims is thus inserting this prior art algorithm into software to perform its  
11 conventional function in a conventional way—i.e., allowing the comparison of the resulting  
12 calculation to a known reference value to detect if software tampering has occurred, and trying to  
13 leave as little evidence as possible that this anti-tampering calculation had been performed. These  
14 "fundamental . . . practice[s] long prevalent" are abstract ideas. *Alice*, 134 S. Ct. at 2356 (citation  
15 omitted). The dependent claims, which relate to storage of the checksum calculation result (claim  
16 2), selection of unspecified checksum algorithms or use of associated information (claims 5, 6, and  
17 30), and performing the algorithm in a step-wise fashion (claims 3 and 31), add nothing more than  
18 generic functions that could be performed by any computer, and thus confirm the abstract nature of  
19 the claimed subject matter. *See, e.g., Clarilogic Inc. v. FormFree Holdings Corp.*, 681 F. App'x  
20 950, 954 (Fed. Cir. 2017) (noting that claims will fail Step 1 of the *Alice* test if there are merely  
21 "directed to a result or effect that itself is the abstract idea and merely invokes generic processes")  
22 (citation omitted).

23 The '149 Patent claims a purely abstract idea that combines a mathematical calculation (a  
24 "checksum") with conventional software programming techniques (removing "checksum-related"  
25 information from the code) and the most rudimentary observational process (determining whether  
26 two values "fail[] to match"). (*See* '149 Patent, claim 1.) This method does not purport to improve  
27 the functioning of a computer, any particular computer component, or hardware, and is not directed  
28

1 to a problem that is "particular to the Internet"; it therefore claims an unpatentable idea. *See Papst*,  
 2 193 F. Supp. 3d at 1081-82.

## 3                   2.       The Asserted Claims Fail The Pen-And-Paper Test

4           In addition to considering whether the Asserted Claims purport to improve computer  
 5 performance or hardware (they do not), and whether they purport to solve an Internet-specific  
 6 problem (they also do not), another "helpful tool" in discerning the nature of claims is the so-called  
 7 "pen-and-paper" test. *Papst*, 193 F. Supp. 3d at 1082. This test asks "whether the claims' steps  
 8 'can be performed in the human mind, or by a human using a pen and paper.'" *OpenTV, Inc. v.*  
 9 *Apple, Inc.*, Case No. 14-cv-01622-HSG, 2015 WL 1535328, at \*4 (N.D. Cal. Apr. 6, 2015)  
 10 (quoting *CyberSource Corp. v. Retail Decisions, Inc.*, 654 F.3d 1366, 1372 (Fed. Cir. 2011)). The  
 11 Asserted Claims unquestionably fail the pen-and-paper test.

12           Claim 1 is representative, and recites essentially four steps: (1) "providing . . . instructions"  
 13 on how to create software that includes checksum-related information; (2) "calcula[ting] at least  
 14 one checksum"; (3) "strip[ping] the checksum information"; and (4) comparing a known checksum  
 15 to the "embedded checksum." (*See* '149 Patent.) Claim 27 is similar, and includes the same basic  
 16 steps of (1) instructing a "software developer" to modify software; (2) "comput[ing] a new  
 17 checksum"; (3) "removing the exported checksum related symbols"; and (4) "compar[ing] the  
 18 computed runtime checksum with an [embedded] checksum." (*See id.*) All of those steps could be  
 19 performed by the human mind or using pen and paper—indeed, the human mind is *necessary* to  
 20 perform the claimed method, because a software developer must conceive of the code that allows  
 21 for the desired result: "determining if protected software is modified." (*Id.*, claim 27.) It would be  
 22 possible to print out computer code onto paper, designate a portion of that code on which to  
 23 perform a checksum calculation, perform that calculation (which just consists of a set of known  
 24 mathematical operations), and compare the result of that calculation to a known value. Even  
 25 though the claims are specific to software, which generally exists in an intangible form, "the pen-  
 26 and-paper test does not require described electronic components to literally exist on paper."  
 27 *OpenTV, Inc.*, 2015 WL 1535328, at \*4. A claim that consists of nothing more than a  
 28 mathematical calculation, instructions to apply general principles of obfuscation, and a comparison



1 that could be done by any person looking at two numbers, is thus a quintessential abstract idea that  
 2 is ineligible for patent protection. *See, e.g., CyberSource Corp. v. Retail Decisions, Inc.*, 654 F.3d  
 3 1366, 1371 (Fed. Cir. 2011) ("[M]ethods which can be performed mentally, or which are the  
 4 equivalent of human mental work are unpatentable abstract ideas . . .").

### 5                   3.       **The Asserted Claims Are Closely Analogous To Claims That Have Been** 6                   **Recently Invalidated Under Section 101**

7           In addition to the facially abstract character of the Asserted Claims, and their failure to  
 8 satisfy the pen-and-paper test, the claims are nearly identical to those that have been routinely  
 9 found to be invalid abstract ideas in the wake of the *Alice* decision. Less than two months ago, the  
 10 Federal Circuit affirmed the 12(b)(6) dismissal of claims that were directed to identifying  
 11 unauthorized computer files because those claims covered an abstract idea with no inventive  
 12 concept. *Intellectual Ventures I LLC v. Erie Indemnity Co.*, No. 2017-1147, 2017 WL 5041460, at  
 13 \*1 (Fed. Cir. Nov. 3, 2017) (to be reported in F. App'x). The patent at issue in *Intellectual*  
 14 *Ventures* was "designed to solve the problem of the proliferation of illicit files on the Internet,"  
 15 including "pirated music or software." 200 F. Supp. 3d at 568. The district court determined that a  
 16 representative claim of this patent "can be boiled down to four fundamental steps: (1) selecting a  
 17 file; (2) **generating a unique value** corresponding to the file; (3) **comparing that unique value** to a  
 18 bunch of previously generated values that correspond to different types of illicit files; and (4)  
 19 marking the file for deletion or other treatment if its assigned value **matches a known one.**" *Id.*  
 20 (emphasis added). The district court then determined that, although the patent claimed a  
 21 "computer-implemented method," that method "does not solve a computer-centric problem or a  
 22 problem that is unique to computers." *Id.* at 574-75. On the contrary, "it merely claims a  
 23 computerized solution to a longstanding problem that exists outside of computers: identifying and  
 24 categorizing illicit files, the possession of which might subject an individual or organization to  
 25 liability." *Id.* at 575. At its heart, this patent was directed to the abstract idea of determining  
 26 whether an identification value "matches another value." The district court concluded that  
 27 "[s]electing files based on identifiers and matching different files/identifiers is just what computers  
 28 do. **There is nothing inventive about it.**" *Id.* at 576 (emphasis added). The Federal Circuit



1 affirmed these findings in their entirety, and further rejected the patentee's argument that the claims  
2 were directed to "an improvement in the way computers operate." *Intellectual Ventures I LLC*,  
3 2017 WL 5041460, at \*4.

4       There are numerous parallels between the invalid patent claims in *Intellectual Ventures* and  
5 those of the '149 Patent. Both patents relate to the problem of software piracy, and both patents  
6 claim a method of identifying illicitly altered computer files. The methodology of the '149 Patent  
7 closely tracks that of the *Intellectual Ventures* patent—both compared a "unique value" calculated  
8 from particular data to "previously generated values." Indeed, the identification value used in the  
9 *Intellectual Ventures* patent is ***exactly the same*** as that in the '149 Patent—both patents use a  
10 "checksum" to determine whether there is a "mismatch," which in turn indicates that a file has been  
11 altered by a computer pirate or hacker. *See* 200 F. Supp. 3d at 576 (noting that there are "numerous  
12 possible algorithms that may be utilized to generate a checksum" in connection with the claimed  
13 invention (citation omitted)). The '149 Patent is thus directed to the same basic problem, using the  
14 same basic method, and the same mathematical calculations as a patent that the Federal Circuit just  
15 weeks ago affirmed was invalid under Section 101.

16       The *Intellectual Ventures* decision is only the most recent in a long line of Federal Circuit  
17 decisions holding that patents claiming algorithms and computer-based comparisons to detect illicit  
18 activity or verify the authenticity of data are unpatentable abstract ideas. As but one example, the  
19 Federal Circuit held that claims directed to filtering spam emails and to using computer virus  
20 screening in a telephone network were ineligible for patenting under Section 101. *Intellectual*  
21 *Ventures I LLC v. Symantec Corp.*, 838 F.3d 1307 (Fed. Cir. 2016) ("*Symantec*"). Like the patent  
22 in *Intellectual Ventures* and the '149 Patent, the first patent in *Symantec* involved "creating file  
23 content IDs using a mathematical algorithm," and then determining whether that mathematically  
24 generated ID "matches" a known "characteristic of other [IDs]." *Id.* at 1313. The Federal Circuit  
25 concluded that this method of "receiving e-mail (and other data file) identifiers, characterizing e-  
26 mail based on the identifiers, and communicating the characterization—in other words, filtering  
27 files/e-mail—is an abstract idea." *Id.* The Federal Circuit reached the same conclusion for another  
28 patent asserted in *Symantec*, which involved "an idea that originated in the computer era—

1 computer virus screening." *Id.* at 1319. Although the patentee claimed to have improved on  
2 known methods of virus detection (just as the '149 Patent purports to improve on known methods  
3 of detecting software modifications), the actual language of the claims was merely drawn to a  
4 "well-known . . . abstract idea." *Id.*

5 Yet further confirmation of the abstract nature of the Asserted Claims can be found in  
6 *FairWarning IP, LLC v. Iatric Systems, Inc.*, 839 F.3d 1089 (Fed. Cir. 2016). In that case, the  
7 claims of the asserted patent encompassed the abstract "concept of analyzing records of human  
8 activity to detect suspicious behavior"—and more particularly, to a method of performing this  
9 analysis for a patient's health records. *Id.* at 1093 (citation omitted). In affirming the invalidity of  
10 those claims, the Federal Circuit noted that it "treat[s] analyzing information by steps people go  
11 through in their minds, or by mathematical algorithms, without more, as essentially mental  
12 processes within the abstract-idea category." *Id.* (citation omitted); *see also Vehicle Intelligence &*  
13 *Safety LLC v. Mercedes-Benz USA, LLC*, 635 F. App'x 914, 915-16 (Fed. Cir. 2015) (affirming  
14 judgment on the pleadings for claims relating to "methods and systems that screen equipment  
15 operators for impairment, selectively test those operators, and control the equipment if an  
16 impairment is detected" because those claims amounted to "only abstract ideas coupled with  
17 routine data-gathering steps and conventional computer activity"); *Planet Bingo, LLC v. VKGS*  
18 *LLC*, 576 F. App'x 1005, 1009 (Fed. Cir. 2014) (affirming that claims were unpatentable under  
19 Section 101 where those claims described a software program "used for the generic functions of  
20 storing, retrieving, and verifying a chosen set of bingo numbers against a winning set of bingo  
21 numbers").

22 Aside from the myriad Federal Circuit cases holding that subject matter analogous to that of  
23 the '149 Patent is patent-ineligible, this Court has frequently made similar determinations. For  
24 instance, in *OpenTV*, this Court granted a motion to dismiss because the claims asserted in that case  
25 were "directed to the abstract idea of compiling, organizing, and transmitting information, using  
26 identification codes as shorthand for that information." 2015 WL 1535328, at \*3. The invalid  
27 *OpenTV* claims are similar in many respects to the Asserted Claims of the '149 Patent. First, they  
28 were each addressed to similar problems—the *OpenTV* patent claims describe a method of

1 preventing a "malicious eavesdropper" from gaining access to communications lines, while the  
 2 '149 Patent claims describe a method of preventing an unscrupulous computer pirate from gaining  
 3 access to software. *Id.* at \*4 (citation omitted). As this Court observed, "[t]he problem of how to  
 4 transmit, receive, store, and organize confidential information deriving from multiple sources is not  
 5 a creature of the Internet age: solutions to this problem date back to the invention of smoke  
 6 signals." *Id.* So too is the problem of informational piracy—the circumventing of defenses against  
 7 those who would copy or tamper with data has long been addressed using mathematical codes or  
 8 checks to try to maintain the integrity of that data and prevent free-riding. Just as the "specific  
 9 solution" to eavesdropping in *OpenTV* "boil[ed] down to the [abstract idea of] us[ing] of  
 10 identification codes to organize and transmit confidential information," *id.* at \*4, the "solution" to  
 11 the problem of software piracy in the '149 Patent "boils down" to using conventional mathematical  
 12 formulas to verify that the code has not been altered.

13 Likewise, in *Blue Spike, LLC v. Google Inc.*, No. 14-cv-01650-YGR, 2015 WL 5260506  
 14 (N.D. Cal. Sept. 8, 2015), this Court granted a motion for judgment on the pleadings after  
 15 concluding that the patents at issue were directed to "computer-based content comparisons" that  
 16 were an unpatentable "abstract idea." *Id.* at \*1. The *Blue Spike* patents generally described  
 17 systems and methods for creating "digital fingerprints" from "various 'signals,'" and then creating a  
 18 reference database from those signals. *Id.* "Thereafter, new signals . . . can be similarly processed,  
 19 [and] the resulting abstract ***checked against the database to determine whether the new signal***  
 20 ***matches*** any earlier analyzed signal." *Id.* (emphasis added). This Court concluded that "the claims  
 21 at issue are generally directed to the abstract concept of comparing one thing to another." *Id.* at \*5.  
 22 The '149 Patent claims are generally directed to the same abstract concept—comparing one  
 23 checksum calculation to a known reference value. *See also Asghari-Kamrani v. United Servs.*  
 24 *Auto. Ass'n*, No. 2:15cv478, 2016 WL 3670804, at \*4 (E.D. Va. July 5, 2016) (granting a motion to  
 25 dismiss infringement claims where the patent-in-suit was merely a combination of two abstract  
 26 ideas: "the use of a third party intermediary to confirm the identity of a participant to a transaction  
 27 and the use of a temporary code to confirm the identity of a participant to a transaction").  
 28

1 The Asserted Claims of the '149 Patent fit squarely within the well-established precedent of  
 2 the Federal Circuit and this Court, both of which have invalidated patent claims that use well-  
 3 established mathematical principles to verify the integrity of information, data, or computer-related  
 4 activities. Based on their plain language and the foregoing precedent, there can be no question that  
 5 the Asserted Claims represent an unpatentable, abstract concept that fails Step 1 of the *Alice* test.

6 **B. The Asserted Claims Fail to Satisfy Step 2 Of The *Alice* Test Because They Do**  
 7 **Not Contain An Inventive Concept**

8 The '149 Patent also fails Step 2 of the *Alice* test. There are no limitations in any of the  
 9 Asserted Claims—whether considered as individual elements or as an ordered combination—that  
 10 add anything beyond the most basic abstract method steps of inserting checksum-related  
 11 information into a software program, performing a mathematical calculation, deleting the  
 12 checksum-related information from the computer code, and comparing the results of the checksum  
 13 to a pre-calculated reference value. This bare functional description falls far short of the  
 14 "significantly more" than an abstract concept that is required to satisfy Step 2 of the '149 Patent,  
 15 because it consists only of "generic" and conventional steps relating to writing software and  
 16 making basic comparisons (albeit through the use of a computer). *Alice*, 134 S. Ct. at 2355, 2357;  
 17 *see also Intellectual Ventures I LLC v. Capital One Fin. Corp.*, 850 F.3d 1332, 1341-42 (Fed. Cir.  
 18 2017) (finding patent claims invalid under *Alice* Step 2 where the claims "merely describe the  
 19 functions of the abstract idea," despite using "technical sounding" terminology); *see also*  
 20 *Ultramercial*, 772 F.3d at 715-16.

21 As noted above, the Asserted Claims do not purport to improve or change the functioning  
 22 of a computer or any computer hardware—and any purported improvement in the functionality of  
 23 software cannot be considered an inventive concept because the claims are merely directed to well-  
 24 known software programming techniques, well-known mathematical algorithms, and well-known  
 25 principles of observational comparison. There is nothing unconventional about the claimed  
 26 "checksum" or its associated "information"—none of the Asserted Claims actually specifies which  
 27 particular checksum algorithm should be used, precisely where checksum information or the  
 28 checksum calculation should be performed, or what that "information" even constitutes. In any

1 event, the '149 Patent itself discloses that there were various code locations for checksums and  
 2 means of inserting checksums that had typically been used to verify the integrity of software in the  
 3 prior art. ('149 Patent at 1:42-47; 3:41-45.) Even if there was something unconventional about  
 4 including the checksum in "executable code," which the '149 Patent specification makes clear is  
 5 not the case, then this would merely be a "limitation," rather than an inventive concept. *See, e.g.,*  
 6 *Ancora Techs., Inc. v. HTC Am., Inc.*, No. C16-1919 RAJ, 2017 WL 6389329, at \*3-5 (W.D.  
 7 Wash. Dec. 14, 2017). In *Ancora*, which was decided less than two weeks ago, the court granted a  
 8 motion to dismiss a patent that, like the '149 Patent, was directed to a method for "reduc[ing]  
 9 software piracy." *Id.* at \*3 (citation omitted). Specifically, the patent in *Ancora* claimed a "method  
 10 of identifying and restricting an unauthorized software program's operation," by inserting a license  
 11 verification "key" into a computer's "Basic Input/Output System ("BIOS")." *Id.* at \*1. This  
 12 allegedly made it "harder for a hacker to tamper with the 'key' when it is stored in the BIOS as  
 13 opposed to other storage areas." *Id.* The *Ancora* court rejected this argument and found that  
 14 "[s]pecifying that the BIOS be used to house the verification structure" was a mere limitation, and  
 15 not something that "transform[s] the [underlying] abstract idea into something patent-eligible." *Id.*  
 16 at \*5.

17 Similarly, obfuscation techniques to thwart hackers like those recited in step (b)(iv) of  
 18 asserted claim 27 were well known at the time the '149 Patent was filed, and the abstract  
 19 obfuscation strategies outlined in the claims do not specify anything beyond the use of standard  
 20 software programming techniques (such as deleting code). (*See id., see also id.* at 3:38-40; 4:1-10.)  
 21 And while the Asserted Claims describe an automated process of comparing values, (*see e.g. id.* at  
 22 14:61-64; 15:6-7), this process is effectively identical to what anyone proofreading or checking a  
 23 document would do in the "brick-and-mortar" context. *Papst*, 193 F. Supp. 3d at 1082 (citation  
 24 omitted. The "stripping" and "comparing" steps in the Asserted Claim thus add "nothing of  
 25 practical significance to the underlying abstract idea." *Ultramercial*, 772 F.3d at 716. In short,  
 26 "there is no other aspect of the [Asserted Claims] that is anything but conventional," which means  
 27 that they necessarily fail Step 2 of the *Alice* analysis. *Symantec*, 838 F.3d at 1319.

Moreover, the Asserted Claims are directed to the sort of purported "improvements" that this Court determined were inadequate to constitute an inventive concept in *Protegrity USA, Inc. v. Netskope, Inc.*, No. 15-cv-02515-YGR, 2015 WL 6126599 (N.D. Cal. Oct. 19, 2015). The asserted claims of the *Protegrity* patent were directed to methods of "detecting intrusion in a database" by determining if the results of a database query "violate the intrusion detection policy," and then restricting access. *Id.* at \*1. In the context of *Alice*'s Step 2, the patentee argued that the *Protegrity* patent-in-suit "improves upon the prior art because it permits access to be restricted *before* information is transmitted to a user where an impermissible request is detected." *Id.* at \*7. This Court rejected that argument, finding that it "does not constitute an inventive concept sufficient to save the claims." *Id.* Similarly, even if the Asserted Claims disclose a method of utilizing checksums in software to verify data integrity that is nominally better in certain circumstances from existing methods of utilizing checksums in software to verify data integrity, that "straightforward idea" is not "significantly more" than the underlying abstract concept. *Id.* at \*4, \*7 (citation omitted). The '149 Patent claims no special technology, and its claims rely entirely on known software programming techniques and standard mathematical calculations from known algorithms to achieve the desired results. Such subject matter does not constitute an inventive concept within the meaning of *Alice*. See *OpenTV*, 2015 WL 1535328, at \*6-7 (determining that claims lacked an inventive concept because they were directed merely to "well-understood, routine, and conventional activities commonly used in industry" (citation omitted)).

As a final point, "in considering whether a claim contains an inventive element, courts often apply the so-called 'machine-or-transformation' test, which asks whether a process is (1) 'tied to a particular machine or apparatus,' or (2) 'transforms a particular article into a different state or thing.'" *Papst*, 193 F. Supp. 3d at 1083 (quoting *Ultramercial*, 772 F.3d at 716). While not dispositive on the question of whether a claim passes muster under Section 101, it is nonetheless a "'useful clue' in the second step of the *Alice* framework." *Id.* The Asserted Claims fail both prongs of the machine-or-transformation test.

First, the Asserted Claims are all method claims that are not tied to a particular machine; no claims recite a particular type of computer, any particular hardware, or other systems. Indeed,

"nowhere does the ['149 Patent] tie the claims" at issue "to a novel machine"—or any machine at all. *Ultramercial*, 772 F.3d at 717. Instead, the language is purely functional—"providing" checksum-related instructions, "calculating" a checksum, "removing" checksum-related information, and "determin[ing]" if a new checksum matches an embedded reference checksum. (See, e.g., claim 1 of the '149 Patent; see also *id.* at 11:37-41 ("[S]oftware written according to the present invention may be stored on a computer-readable medium, such as a removable memory, or transmitted over a network, and loaded into a computer for execution.").) Thus, even if the Asserted Claims require the use of a generic computer, that is inadequate to provide an inventive concept. See, e.g., *Bancorp Servs., L.L.C. v. Sun Life Assurance Co. of Canada (U.S.)*, 687 F.3d 1266, 1278 (Fed. Cir. 2012) ("[T]he use of a computer in an otherwise patent-ineligible process for no more than its most basic function—making calculations or computations—fails to circumvent the prohibition against patenting abstract ideas and mental processes."); see also *CyberSource*, 654 F.3d at 1376 ("[M]erely claiming a software implementation of a purely mental process that could otherwise be performed without the use of a computer does not satisfy the machine prong of the machine-or-transformation test."). The Asserted Claims are devoid of any structural elements, let alone any specific machine or computer system and thus fail the first prong of the machine-or-transformation test.

Second, the Asserted Claims do not transform anything into a different state or thing. The Asserted Claims both begin and end with the same intangible matter: computer software. The only alteration of that software is to insert similarly intangible matter—an algorithm for calculating a checksum—which cannot be considered "transformative." On the contrary, the goal of this methodology, according to the specification of the '149 Patent, is to make it appear that the software has not been changed at all, so as not to tip off potential hackers to the presence of the checksum calculation. ('149 Patent at 4:11-15.) There is no transformation that satisfies the machine-or-transformation test in these claims, which begin and end with computer code.

## VI. CONCLUSION

For the foregoing reasons, the claims of the '149 Patent are directed exclusively to unpatentable subject matter and are invalid under 35 U.S.C. § 101. Plaintiff has thus failed to state

1 a claim for infringement, and the Complaint against Defendants should be dismissed in its entirety  
2 and with prejudice.

3  
4  
5 DATED: December 26, 2017

Respectfully submitted,

6 SKADDEN, ARPS, SLATE, MEAGHER & FLOM LLP

7 By: /s/ James Y. Pak

8 James Y. Pak

9 Attorney for Defendants  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28